RL-TR-96-178
Final Technical Report
December 1996

REGION-OF-INTEREST CORRELATION FILTERS

University of Dayton Research Institute

Sponsored by
Advanced Research Projects Agency
ARPA Order No. C931

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

Rome Laboratory
Air Force Materiel Command
Rome, New York

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RL-TR-96-178 has been reviewed and is approved for publication.

APPROVED:

JAMES L. DAVIS
Project Engineer

FOR THE COMMANDER:

GARY D. BARMORE, Major, USAF
Deputy Director
Surveillance & Photonics Directorate

If your address has changed or if you wish to be removed from the Rome Laboratory mailing list, or if the addressee is no longer employed by your organization, please notify RL/OCPC, 25 Electronic Parkway, Rome, NY 13441-4514. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REGION-OF-INTEREST CORRELATION FILTERS

Contractor: University of Dayton Research Institute
Contract Number: F30602-95-1-0038
Effective Date of Contract: 1 January 1995
Contract Expiration Date: 15 May 1996
Program Code Number: 5Y10
Short Title of Work: Region-of-Interest Correlation Filters

Period of Work Covered: Jan 95 - May 96

Principal Investigator: David L. Flannery
Phone: (513) 229-3975

RL Project Engineer: James L. Davis
Phone: (315) 330-4276

Approved for public release; distribution unlimited.

This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by James L. Davis, Rome Laboratory/OCPC, 25 Electronic Pky Rome, NY 13441-4514.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1996	Final Aug 95 - May 96

4. TITLE AND SUBTITLE

REGION-OF-INTEREST CORRELATION FILTERS

5. FUNDING NUMBERS

C - F30602-95-1-0038
PE - 62712E
PR - C931
TA - 00
WU - 01

6. AUTHOR(S)

David L. Flannery

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Dayton Research Institute
300 College Park
Dayton OH 45469-0150

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington VA 22203-1714

Rome Laboratory/OCPC
25 Electronic Parkway
Rome NY 13441-4515

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

RL-TR-96-178

11. SUPPLEMENTARY NOTES

Rome Laboratory Project Engineer: James Davis/OCPC/(315) 330-4276

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This effort investigated Region-of-Interest (ROI) filter formulations for Binary Phase-Only Filter (BPOF) implementation in coherent optical correlator systems. A Region-of-Interest may be broadly defined as a localized region in a sensor pattern (an image) that has a high probability of containing a pattern (a target image) of interest for an application such as automatic target recognition (ATR). Such filters are useful in the early stages of an ATR process because they can limit the total area of an input image that is processed, thus reducing the required processor throughput (and size, weight, and power consumption). Few if any formal procedures for designing ROI correlation filters have existed prior to this effort, while formulations for more specific filters matched to a narrow range of targets and target aspects are relatively numerous. The work reported here developed and applied concepts of normalized BPOF distance, cluster analysis, and the least aggregate square error (LASE) correlation filter formulation to yield a successful ROI filter formulation procedure, called the cluster-based ROI BPOF. The formulation was implemented in software suitable for use on desktop computers, and the validity of the resulting ROI filters was demonstrated on a realistic test problem.

14. SUBJECT TERMS

Optical correlation, correlation filter, cluster analysis

15. NUMBER OF PAGES 40

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

Table of Contents

REGION-OF-INTEREST CORRELATION FILTERS

1.0 INTRODUCTION

This is the final technical report for an effort performed by the University of Dayton Research Institute under Grant F30602-95-1-0038 for the Air Force Rome Laboratory (AFMC), Rome, New York. The reported effort involved study, analysis, software development, and computer simulations of optical correlation and it was performed in the Electrical and Computer Engineering Division of the Research Institute headed by Mr. Dennis D. Stafford. The principal investigator was Dr. David L. Flannery and the effort was performed over the period from December 1995 through May 1996.

This effort investigated region-of-interest (ROI) filter formulations for binary phase-only filter (BPOF) implementation in coherent optical correlator systems such as the SPOTR (System for Passive Optical Target Recognition) developed by Lockheed Martin for the ARPA TOPS (Transfer of Optical Processing to Systems) program [1]. A region-of-interest may be broadly defined as a localized region in a sensor pattern (e.g., an image) that has a high probability of containing a pattern (e.g., a target image) of interest for an application (e.g., for automatic target recognition, ATR). Such filters are useful in the early stages of an ATR process because they can limit the total area of an input image that is processed, thus reducing the required processor throughput (and size, weight, and power consumption). Few if any formal procedures for designing ROI correlation filters have existed prior to this effort, while formulations for more specific filters matched to a narrow range of targets and/or target aspects are relatively numerous.

The work reported here developed and applied concepts of normalized BPOF distance, cluster analysis, and the LASE correlation filter formulation [2] to yield a successful ROI filter formulation procedure, called the cluster-based ROI BPOF. The formulation was implemented in software suitable for use on desktop computers, and the validity of the resulting ROI filters was demonstrated on a realistic test problem.

Section 2 provides a summary of the effort and its results, including an overview of the new filter design process. Section 3 presents BPOF distance concepts. Section 4 describes the cluster analysis procedure developed and its implementation in software. Section 5 describes the software developed to formulate the ROI filters based on the cluster analysis. Section 6 presents the test problem and results. Section 7 provides concluding remarks and recommendations for further study.

2.0 SUMMARY OF THE EFFORT

The key concepts of the new cluster-based ROI filter are normalized BPOF distance metrics, the clustering algorithm, and the iterated-LASE filter formulation. The filter formulation process uses these concepts sequentially in the order mentioned.

2.1 Normalized BPOF distance metrics

Classical correlation using true matched filters can be directly related to a Euclidean distance metric involving the integration of the product of two pattern functions over their domain. If the two patterns are energy normalized, the distance between them falls in the range (0,1), with the value zero occurring only if the normalized patterns are identical. Thus the distance metric is a measure of the dissimilarity of two patterns.

The Euclidean distance cannot be related directly to correlation of a function with a BPOF. However, new distance metrics can be defined in terms of BPOF correlations. (See Section 3 for details.) Two new BPOF-related normalized distance metrics were defined, one symmetrical and one non-symmetric. The symmetric definition is useful when a general BPOF distance is required and neither pattern is more likely to be encoded in a BPOF than the other. The non-symmetric version is used when evaluating distances between a particular pattern encoded in a BPOF and other patterns.

The defined normalized BPOF distance metrics have the useful property that they are zero only if the two patterns have the same BPOF, thus yielding the maximum possible BPOF correlation for those patterns.

These distance metrics provide the basis for clustering in BPOF space and for iterative design of LASE filters to provide improved uniformity of response across a training set.

2.2 Cluster Analysis

A C language program was created to generate a symmetric normalized BPOF distance matrix for a set of training images representing target poses spanning a distortion hyperspace. A clustering algorithm was designed to find at least M clusters that maximize inter-cluster distances while minimizing intra-cluster distances. The minimum number of clusters, M, is operator specified. The only input to the clustering algorithm is the value M and the distance matrix.

The cluster algorithm was implemented in a C language program called cluster.c. The program outputs the cluster information.

2.3 Cluster-based Filter Generation

A C language program, roifilt.c, to generate ROI filters based on the training image groups defined by the cluster analysis was created. It is a modification of one of the many variants of "filtermeister" programs created in efforts sponsored by Lockheed Martin. The cluster poses are encoded in the BPOF using the LASE filter formulation [2]. For this effort, a new extension of the LASE formulation, called iterated-LASE (or ILASE) was developed. The LASE formulation satisfies minimum-square-error goals, but it does not address uniformity of response over a training set. The ILASE formulation performs iterations to improve the response uniformity while still satisfying the original LASE goals.

2.4 ROI Filter Design Test Case

A test case was defined using EOTADS IR images supplied by Lockheed Martin and the U.S. Army Night Vision Laboratory (NVL). The target is an M60 tank. The set included training images covering 360 degrees of azimuth and a number of cluttered scenes containing the M60.

First, 120 poses were defined from the training images using 15-degree azimuth intervals over 360 degrees and five range steps spanning 2500M to 4000M range. The corresponding 120-by-120 element distance matrix was computed using distmtrx.c. Then cluster.c was used to perform cluster analysis to define four clusters. Roifilt.c was then used to generate four cluster-based ROI BPOF's. The total computation time on a desktop computer was only a few minutes.

The filters were tested in correlation simulations using 12 input scenes. The target's region of interest was found as the first- or second-ranked correlation peak in 77% of the 48 correlations. The same input scenes were tested with previously investigated ROI filter types based on disk, ring, and elliptical reference shapes. The best of these provided only 25% correct ROI detection in the first- and second-ranked correlation peaks. Thus the superiority of the cluster-based ROI BPOF is convincingly demonstrated in this test case.

The reader is referred to Section 7 for concluding remarks and recommendations for further study.

3.0 BPOF DISTANCE METRICS

3.1 Review and Problem Definition

A normalized distance metric applicable to BPOF correlation is a critical element of the cluster-based ROI filter formulation developed in this effort. Cluster analysis is applied with the goal of improving filter banks by building each filter to cover the smallest possible distortion space while still covering the entire distortion space with a specified number of filters. If training images (however derived) are subjected to cluster analysis, it should be possible to regularize the distortion coverage across a filter bank. Thus, for example, one smart filter might "cover" 100 training images while another might cover only 10 training images, based on the finding that each training image cluster actually extends over the same size region in the space of a suitable distance metric, i.e., cluster analysis indicates both training image subsets require approximately the same distortion tolerance of their respective filters. Since there is a well known fundamental trade-off between distortion tolerance and other important correlation performance parameters (e.g., clutter rejection), it is important to ask each filter to cover the same ***true*** distortion range (as measured by an appropriate distance metric). Otherwise the filter bank will exhibit wide variations of correlation performance between filters, which are very difficult to deal with in a filter strategy.

The (classical) cross-correlation between two patterns can be chosen as the basis for a Euclidean distance metric. The cross-correlation value may be expressed in Fourier space, using Parseval's theorem, as:

$$L_{f,g} = \int F^*(k)G(k)dk \quad , \tag{1}$$

Where L is the cross-correlation value, and $F(k)$, $G(k)$ are the Fourier transforms of two real patterns $f(x)$, $g(x)$. The variables x and k are suitably dimensioned space and spatial-frequency pairs. The integral may be recognized as the zero-shift value of the cross-correlation *function* of f and g. The Euclidean distance chosen for this example is:

$$D_{f,g} = \left(\int [f(x) - g(x)]^2 dx\right)^{\frac{1}{2}}, \tag{2}$$

or, squaring both sides and expanding the integration kernel:

$$D_{f,g}^2 = \int f(x)^2 dx + \int g(x)^2 dx - 2\int f(x)g(x)dx = \int |F(k)|^2 dk + \int |G(k)|^2 dk - 2\int F^*(k)G(k)dk \quad , \tag{3}$$

where Parseval's theorem was again used to obtain the final Fourier-space form of the expression. The first two terms above are the energies in the two patterns, which may be normalized to unity as an implicit part of the pattern's

original definition. The third term is twice the cross-correlation value. Substituting Eq. 1 into Eq. 3 yields:

$$D_{f,g} = \sqrt{2\left(1 - L_{f,g}\right)} \quad , \tag{4}$$

where only the positive root is used. This distance metric has a maximum value of $\sqrt{2}$ when f and g are completely orthogonal (i.e., have a cross-correlation value of zero) and a minimum value of zero when f and g are identical (i.e., perfectly correlated).

The relationship between the distance metric and the cross-correlation value is inverted and monotonic, i.e., the greater the distance the smaller the cross-correlation. This is hardly surprising since cross-correlation is a well known measure of similarity. Thus distance criteria of the form $D < D_0$ are equivalent to $L > L_0$, where D_0 and L_0 are related by Eq. 4.

The distance metric defined by Eq. 4 may be used for cluster analysis based on *classical* matched filtering, in which Eq. 1 is by definition the on-axis (x=0) correlation peak value. ($F^*(k)$ is the matched filter while $G(k)$ is the Fourier transform of the input pattern.) However, Eq. 1 does not apply to phase-only filtering or to binary phase-only filtering. The BPOF for function $f(x)$ is:

$$BPOF_f(k) = \mathrm{Sign}\left[\mathrm{Re}\left(e^{i\phi} F(k)\right)\right], \tag{5}$$

where φ is the threshold line angle (TLA), a parameter of the BPOF.

The correlation magnitude obtained by BPOF correlation is a modified version of Eq. 1:

$$L_{f,g}^{BPOF} = \left| \int BPOF_f(k) G(k) dk \right| . \tag{6}$$

A distance metric for BPOF correlation could be defined as in Eq. 4 using this modified similarity measure. However this is undesirable because such a definition would be non-symmetric, i.e., $L_{f,g}^{BPOF} \neq L_{g,f}^{BPOF}$. This lack of symmetry corresponds to the fact that BPOF correlation results depend on which function (f or g) defines the BPOF. If we are searching for desirable filter-defining clusters in a large set of training images, there is no basis for selecting which of two non-symmetric distance values should be used to define the distance between a pair of patterns. An additional complication is that energy normalization, as used for the classical Euclidean distance concept, is not valid for BPOF correlation; i.e., the BPOF auto-correlation is not constrained to unity by using energy normalized functions.

3.2 BPOF Normalization

The BPOF normalization condition is derived simply by constraining the BPOF auto-correlation (actually the central value of the auto-correlation function) to unity:

$$\left|\int BPOF_f(k)F_n(k)dk\right| = 1, \tag{7}$$

where F_n denotes the normalized F. Thus, by definition:

$$F_n(k) = \frac{F(k)}{\left|\int BPOF_f(k)F(k)dk\right|} \quad . \tag{8}$$

Thus the normalized Fourier transform may be computed from the unnormalized transform by dividing by the appropriate normalizing constant, which is the denominator of Eq. 8. Note this normalization also can be carried out in the function domain by simply dividing by the same normalization constant. Note also that the TLA, implicit in the $BPOF$ function, is a parameter of this normalization.

Hereafter, the discussion assumes that BPOF-normalized functions are used, and the explicit n subscript is dropped.

If normalized transforms are used, the similarity measure defined in Eq. 6 has a maximum value of unity for BPOF auto-correlation. However, because the BPOF is not uniquely related to the transform from which it is computed (depending only on the phase of the transform), there are in principle an infinity of functions other than *f*, whose BPOF's will produce unity magnitude when correlated with *f*. The BPOF of any function whose Fourier transform phase matches the phase of F or, less restrictively, whose transform maps to the same BPOF function will produce the maximum magnitude of unity. Thus the powerful statement that applies to normalized classical correlation, i.e., that only functions identical to within a normalization constant yield the maximum (unity) normalized correlation, does not apply to normalized BPOF correlation. As a practical matter, one must ask what the probability is that two functions will have identical BPOF functions. This uniqueness condition is not critical to further development of the cluster-based filter formulation.

3.3 Normalized Distance Metrics

A non-symmetric normalized distance metric can be defined simply by:

$$D_{f,g}^{BPOF} = 1 - L_{f,g}^{BPOF} = 1 - \left|\int BPOF_f(k)G(k)dk\right| \quad , \tag{9}$$

noting that the BPOF-normalized functions are implicitly denoted. This function is bounded by 0 and 1, and becomes zero when $f = g$. Obviously variations on this definition are possible. For example the form:

$$D_{f,g}^{BPOF} = \sqrt{1-\left(L_{f,g}^{BPOF}\right)^2} \tag{10}$$

could be chosen. However advantages, if any, of the more complex form are unknown and it requires more computation, an important factor when computing the numerous terms in a distance matrix. Either form satisfies the requirement for an inverse monotonic relation between the distance and similarity measures.

A symmetric distance metric is highly desirable for cluster analysis, as mentioned. In this effort a symmetric distance was synthesized simply as follows:

$$D_{f,g}^{sym} = D_{g,f}^{sym} = \frac{\left(D_{f,g}^{BPOF} + D_{g,f}^{BPOF}\right)}{2} \quad , \tag{11}$$

i.e., as the average of the two non-symmetric distances. This distance clearly is bounded identically as the non-symmetric distance and also is zero when the two functions are identical.

Note that the non-symmetric distance is ideal for use in filter design algorithms, such as the ILASE formulation to be discussed in Section 5, where a trial BPOF has been defined and a measure of its proximity or distance to a set of training images is to be computed.

3.4 Software for Distance Matrix Generation - Distmtrx.c

Distmtrx.c computes all the mutual normalized BPOF distances for a set of images specified by an input list file, i.e., the entries for a distance matrix. The distances are output in a text file formatted to serve as an input file to program cluster.c, which performs a cluster analysis.

Distmtrx operation is controlled entirely by a single text file that provides all the run-time parameters. A training set of images must exist on a specified computer subdirectory. These images normally cover variations in azimuth, and elevation if applicable. In-plane distortions consisting of rotation (called tilt) and scale (or range) are generated digitally during program operation.

The list format is as follows:

```
distmtrx-list
slmsize [128 or 256]
sensor type [RSS, LCIR, TRIM, EOTADS, MIS]
target type [M1, M60, T72, 2S1, M2, M113, BMP, M35, HMMWV, XXXX(MIS)]
reference type [ hot, cold, fused, visible, model]
preprocess type [range2, modrange2, range3, modrange3, histo, freichen,frei-
edge,sobel, sobel-edge, none]
start azimuth(image#) 0
end azimuth(image#) 348
azimuth(image#) increment 12
start range(scale) 1500
end range(scale) 500
scale steps 15
start elevation 6
end elevation 6
elevation increment 3
start tilt 0
end tilt 0
tilt increment 2
TLA 30
Quantize "off" (or) mode# #bits [lower_% [upper_%] ]
binarization percentage 50
output_datafile thiscase.dat
reference directory /disk3/references/eotads/t72/
log file distmtrx.log
```

The program uses the azimuth, range, elevation, and tilt parameters to generate a set of poses covering the desired distortion hyperspace. Then it computes the symmetrical normalized BPOF distances between each pair of poses and outputs the results. Other list parameters that prescribe preprocessing options are not described here but they have been described previously [3].

A log file is generated but the primary output is written to the specified text output_datafile name. The first line of the output file is an integer equal to the

number of distinct poses, call it N. Then the distortion parameters for the N poses are listed in N lines. The format of this listing is:

Pose# azimuth elevation range tilt

where pose# is a sequential index assigned by the program.

The remaining N*(N+1)/2 lines of the output file list the BPOF distances with the following format:

pose_a# pose_b# BPOF-distance

These data comprise the diagonal and upper-triangular entries for the (symmetric) distance matrix. The diagonal entries serve as a consistency check as these distances must by definition be zero.

4.0 CLUSTER ANALYSIS

Cluster analysis is performed by a C language computer program, cluster.c. The only inputs to the process are the distance matrix, computed as described in the previous section, and an operator specified minimum number of clusters to be defined, M.

4.1 Clustering Algorithm

Let MinC be the specified minimum number of clusters to be defined. The clustering goal is defined as follows:

Find the maximum (normalized BPOF) distance threshold (DThresh) for which at least MinC poses can be found that are mutually at least DThresh distant. These poses are called cardinal poses. Each one serves as the agglomeration point for a cluster. After finding these cardinal poses, clustering is accomplished by simply assigning each of the remaining poses to the cluster associated with its nearest cardinal pose.

An outline of the algorithm for finding a good DThresh is given here. Note that normalized BPOF distance values all fall in the range (0, 1).

1. An initial value of DThresh is chosen. It must be high enough to ensure that it is too high to yield MinC cardinal poses. The program currently starts with DThresh = 0.96 .
2. An array is used to keep a list of poses that are still candidates as cardinal poses. This list is initialized to include all N poses.
3. The entire set of poses is scanned and a "nearness score" is assigned to each pose. For each other pose that is closer than (DThresh - 0.2), the pose' score is incremented by three. For each pose within (DThresh - 0.1) the score is incremented by two. For each pose within DThresh the score is incremented by one. The maximum score over all poses is tracked.
4. All poses scoring the maximum nearness from step 3. are removed from the cardinal pose list.
5. Steps 3. and 4. are repeated until the maximum score is zero or until no poses remain on the cardinal pose list.
6. If the number of poses on the cardinal pose list is less than the required minimum, DThresh is decreased by a fixed increment (currently 0.04 in the program) and steps 2. through 5. are repeated.

Non-cardinal poses are assigned to clusters associated with the cardinal poses. The nearest cardinal pose defines the cluster assignment for each pose. The

RMS value of the mutual distances within each cluster are computed and output to the screen.

4.2 Clustering Software Implementation - Cluster.c

Cluster.c implements the clustering algorithm just described to perform training set clustering based on symmetrical normalized BPOF distances generated by program distmtrx.c. Its output file serves as an input file to program roifilt.c, (see Section 6) which is a filtermeister-like program that generates the iterated-LASE filters corresponding to the clusters defined by cluster.c. It is invoked in a DOS command line as follows:

cluster input_file   output_file   [min. # of clusters]

The text input file is generated by distmtrx.c, (see Section 4) and it contains the distance matrix values and pose information. A properly formatted input file could be synthesized from other sources if desired.

The text output file contains the cluster information generated by the program and is formatted for input to roifilt.c.

If the optional minimum number of clusters is not specified on the command line, a default value equal to one tenth of the number of poses is used.

The program operates iteratively, with a trend toward generating more clusters on each iteration. The first iteration to generate at least the minimum number of clusters will cause termination. Thus the actual number of defined clusters may be greater than the minimum.

The program generates some screen output, including the RMS distances associated with each cluster defined. This output can be re-directed to a file for logging if desired. The primary output is written to the output file specified as a command-line parameter.

The output file format is:

First, three lines as follows:

(Number of poses, N)

(Number of clusters defined, NC)

(Maximum number of poses in a cluster)

Then (NC+N) lines in the following format:

Cardinal_pose# azimuth range

Of these, the first NC lines define the cardinal poses determined by the program. The remaining N lines are a sequential listing by pose number (an index defined by the order of poses in the input data file) in which the first number on each line is the cardinal pose for the cluster to which that pose has been assigned.

NOTE: The elevation and tilt parameters present in the input file are ignored by cluster.c because they are not currently handled by the smart filter generator program (roifilt.c) that uses the output file generated by cluster.c. Cluster.c could easily be modified to read these values and include them in its output file.

5.0 SMART FILTER GENERATION BASED ON CLUSTERS

The training set, or pose, clusters generated as described in the previous two sections are used as inputs to a computer program that generates an ILASE filter for each cluster.

5.1 LASE Filter Formulation

The LASE (least aggregate square error, [2]) filter formulation meets the following defining goal:

Find the single filter, subject to constrained modulation, that minimizes the aggregate square-error in the correlation functions over a set of N test images consisting of a set of N training images with additive random noise described by a known power spectral density. The "error" is defined as the difference in (complex) correlation amplitude between correlations of a particular input with a filter optimized solely for that input and with the single LASE filter. The individual optimized filters may be optimized in any desired manner. The aggregate square-error is the magnitude-squared of this difference summed over the entire correlation plane and over the set of test images. In different words, the LASE filter is the single compromise filter that most closely approaches the performance of the N individual filters in a mean-square error sense, for additive random input noise of a given power spectral density. The defining publication [2] relates the LASE formulation to other published formulations and discusses how the aggregate square error of the filter can vary depending on degeneracies implicit in the optimization of the individual optimized filters. These details are beyond the scope of this report and the reader is referred to the cited reference for further discussion.

The mathematical expression for the LASE filter is:

$$H_{LASE,k} = \mathrm{MED}\left\{ \frac{\sum_{n=1}^{N} H_{n,k} \left| T_{n,k} \right|^2}{\sum_{n=1}^{N} \left| T_{n,k} \right|^2} \right\} , \qquad (12),$$

Where $H_{LASE,k}$ is the LASE filter function value for the k'th spatial frequency, MED denotes a minimum-Euclidean-distance mapping operation, $H_{n,k}$ is the n'th individual optimized filter value for the k'th spatial frequency, and $\left|T_{n,k}\right|^2$ is the power spectrum of the n'th test image. The MED operation outputs the modulation value on the locus of realizable filter modulation in the complex plane that is closest to the argument value.

Note that the LASE formulation is not inherently iterative, which leads to a computation-time advantage over the Jared-and-Ennis formulation that was used in the TOPS program.

In this and prior efforts the modulation constraint was binary-phase modulation. The individual optimized filters were BPOF's based on the binarized training images with an assumed TLA (threshold line angle, [4]) that was known empirically to be near-optimum in terms of combined P/C and peak-intensity performance. Training image power spectrums are the sum of the power spectrum of each training image and an estimated background-noise power spectral density. The background-noise power spectrum estimates may be formed by:

1. Selecting a set of representative input scenes.
2. Applying the appropriate binarization preprocessing to each scene.
3. Editing the target signature out of each binarized scene.
4. Fourier transforming the edited scenes.
5. Computing the magnitude-squared (power spectrum) of each transform.
6. Computing the average power spectrum over the selected set.
7. Polar smearing of the composite spectrum over ±0.05 radian azimuthally and ±5% radially, to reduce alignment sensitivity to straight-line background features.

For the cases reported herein, the background-noise spectrum was assumed to be zero; i.e., the test images were identically the training images. It is possible that ROI performance could be improved by incorporating a representative background noise power spectrum.

The MED operation to form the LASE filter is simply a thresholding of the argument value (which is pure real in this case) at zero to set 1 or -1 filter values.

5.2 The Iterated LASE Filter Formulation

The LASE formulation does not apply constraints to force or encourage response uniformity over the training set. In previous smart filter designs, the response uniformity of LASE filters was found to be acceptable [3, 5]. However in other applications, including ROI filters, it may be desirable to improve the response uniformity of LASE filters. The LASE filter formulation algorithm has been augmented to incorporate iterative procedures for this purpose.

The formulation is modified by introducing real relative weighting coefficients, c_n, as follows:

$$H_{LASE,k} = \mathrm{MED}\left\{ \frac{\sum_{n=1}^{N} c_n H_{n,k} \left| T_{n,k} \right|^2}{\sum_{n=1}^{N} \left| T_{n,k} \right|^2} \right\} . \tag{13}$$

For phase-only filters, including the BPOF of concern herein, only the relative values of the c_n, matter since multiplying all of them by a (real, non-zero) constant cannot change the resulting filter values.

The ILASE formulation builds an initial LASE filter with uniform weights (equal to unity). Then the central correlation intensity of each training image for the current filter is evaluated and the relative weighting of each image is adjusted up or down according to whether its relative BPOF-normalized correlation response is less than or greater than the average BPOF-normalized correlation response over the training set. This adjustment cycle is repeated until termination conditions (e.g., a desired uniformity) are met. The termination condition used in the roifilt.c program reported herein was that the non-symmetric BPOF-normalized correlation distances over the training set be uniform within ±5% or that 20 iterations had been performed.

This relaxation algorithm is very similar to the one first proposed by Jared and Ennis [6] for improving response uniformity of composite phase-only filters across a training set. Note however that the LASE filter formulation, with or without the relative weighting concept, is fundamentally different from a composite filter as discussed in the defining publication [2].

5.3 Filter Generation Software Implementation - Roifilt.c

Roifilt.c generates iterated-LASE filters based on training set (i.e., pose) clusters defined by program cluster.c. The pose parameters and cluster information are contained in an input text file, which normally would be generated by running cluster.c. The program is controlled by a text file with the following format:

```
roi-list
cluster_file_name cluster.txt
slmsize [128 or 256]
filter type [ itl, flc, binpack ]
background spectrum filename
sensor type [RSS, LCIR, TRIM, EOTADS, MIS]
target type [M1, M60, T72, 2S1, M2, M113, BMP, M35, HMMWV, XXXX(MIS)]
reference type [ hot, cold, fused, visible, model] preprocess type [range2, modrange2,
range3, modrange3, histo,freichen, frei-edge, sobel, sobel-edge, none]
TLA 30
Quantize "off" (or) mode# #bits [lower_% [upper_%] ]
binarization percentage 50
listname filter.lst
reference directory /disk3/references/eotads/t72/
log file lase.log
```

List item cluster_file_name is the input data file as generated by cluster.c.

The format of the input data file is specified in Section 4. It includes the specification of N poses, with each pose (currently) defined only by azimuth and range parameters. (Elevation and/or tilt parameters could be incorporated later.) The input file also specifies a number of "cardinal poses", each of which was the agglomeration point for a cluster, and the assignment of each of the N poses to a cluster (i.e., to a cardinal pose).

Upon initialization, the program establishes memory array requirements based on the maximum number of poses in a cluster, a parameter which is read from the input data file. If this maximum exceeds a predetermined value (max_chunk), the program terminates in an error condition. The allowable number of poses in memory is based on a constant (#define MAX_CHUNK) but is dynamically adjusted based on the FFT array size involved.

The program then steps through each cardinal pose (i.e., each cluster), building an ILASE BPOF using the training images (poses) associated with that cluster.

Program outputs include filters, a filterlist, and a log file. Filter names include only the target name and the cardinal pose number for the associated cluster.

6.0 EOTADS TEST CASES

Using EOTADS imagery already on hand as a result of a previous program [7], a realistic test case was defined. A training set of 120 target poses was defined covering 360 degrees of azimuth and ranges of 2500 M to 4000 M. Cluster-based ROI BPOF filters were formulated based on clusters defined by using the software described above, and the filters were tested using 256x256-pixel EOTADS sensor images. The ROI performance of the new filters was compared with that of previously defined ROI filters.

6.1 Test Images

The EOTADS image data sets are described in an NVL document entitled "Standard Analysis of Target Recognizers, Part I: Baselining Test Plan." The EOTADS digital images are 720x480-pixels with 10-bit words for each pixel. The pixels are double-sampled horizontally so 2X horizontal averaging was performed and the images are then treated as 360x480-pixel images. Each image contains only one instance of the selected M60 tank target. A 256X256-pixel window containing the target was selected for the reported tests. Available software was used to perform Sobel preprocessing and to perform the histogram-based (percent threshold) binarization with 10-bit precision (matching the raw data precision). All the cases reported herein used input images thresholded so 10% of the binary pixels were turned on. Target ranges of approximately 2000M and 3400M are available in the EOTADS set but only the longer range images were used in the reported effort

A test set including 12 instances of the M60 at 45°, 180°, and 270° azimuths was prepared. The set was about equally distributed over the three azimuths. . Figure 1 is an example of an input image in both gray scale and binary versions. Note the considerable clutter in this scene, which is about average for the set.

6.2 Training Images

M60 Training images spanning 360 degrees of azimuth at five degree intervals were furnished by Lockheed Martin. The target scale in these images is 2.628 pixels/foot. Two sets of training images, hot- and cold-target, were furnished. Only the hot set was used in the work reported herein. Filter references were prepared by applying Sobel edge enhancement and thresholding so 50% of the target pixels were turned on in the binary images. Range variations were simulated by digital scaling performed by the roifilt.c program.

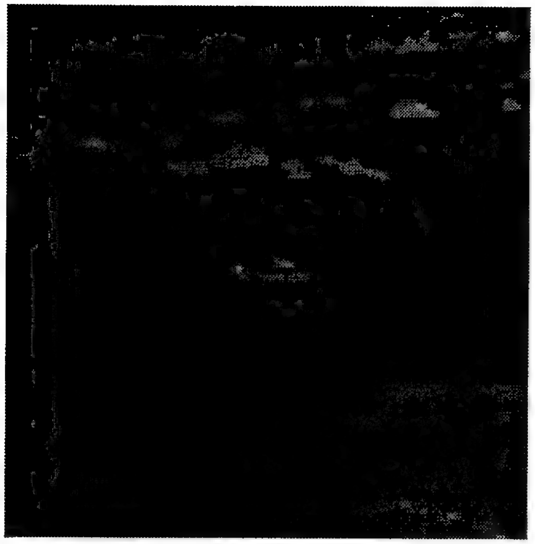 

Figure 1. Example 256x256-pixel input test scene. Gray scale (left) and binary (right) versions. The M60 tank is at 270 degrees azimuth.

6.3 Distance Matrix Generation

A set of 120 poses was defined by selecting training images for azimuths at 30 degree intervals over 360 degrees, and forming five scaled versions at each azimuth to span the 2500M - 4000M range in equal steps. Program distmtrx.c was then used to generate the 14,280 elements of the symmetric distance matrix. An FFT size of 128x128-elements was used.

6.4 Results with Four Clusters

Cluster analysis based on the 120x120-element distance matrix was first performed with the goal of producing a minimum of four clusters. Four clusters were defined and the number of poses in a cluster varied from 14 to 46. The RMS intra-cluster symmetric normalized-BPOF distances ranged from 0.67 to 0.74. The output file was used as input to program roifilt.c , which generated the corresponding four ROI filters. Correlation simulations at 256x256-pixel resolution were performed for every pair defined by the four ROI filters and the 12 test input scenes.

The primary method used to assess ROI performance was a probability of detection (PD) metric that is an estimate of the average probability that an ROI filter will yield a correlation peak in the M top-ranked peaks for each correlation that identifies (by its location) the target ROI. A rectangular box was defined roughly enclosing the target in each of the input scenes. To score a correct ROI identification, a peak had to fall within the box. The choice of M involves an application-specific trade-off between probability of detection and the false-alarm

rate. For this effort the PD was computed for M= 1,2 and 3, which was judged a reasonable range.

A previous effort defined and tested ROI filters based on reference images consisting of simple geometric shapes (discs, rings, and ellipsoids) of various dimensions [3]. To provide a basis of comparison, a large set of these types of ROI filters were also correlated with the 12 test input images. The set included:

1. Solid disk references with diameters from 6 to 12 pixels,
2. Two-pixel thick circular ring references with diameters from 6 to 12 pixels,
3. Ellipsoids with dimensions chosen to approximately match the size of the broadside target (e.g., 30x14 pixels).

The PD results for the cluster-based filters and for the best other type of ROI filter are tabulated:

	M=1	M=2	M=3
Four cluster ROI's	65%	77%	81%
Best Other (30x14 Ellipsoid)	25%	25%	33%

The peak intensities and clutter rejection capabilities of the filters are also of interest. Peak intensities are measured in "units." A unit is the intensity corresponding to one binary pixel turned on in the input plane. The cluster-ROI's averaged 4.73 units of intensity for correctly located ROI peaks and the highest clutter peaks averaged 3.28 dB below the ROI peaks. The corresponding values for the elliptical ROI were 4.70 units and 0.70 dB.

6.5 Results with More Then Four Clusters

The clustering program was re-run to generate cluster sets having 7 and 13 clusters, the corresponding filters were generated, and correlations with the 12 test inputs were performed and analyzed. The PD results for these and the previous (four-cluster) cases are tabulated.

Number of Clusters	M=1	M=2	M=3
4	65%	77%	81%
7	54%	71%	76%
13	62%	76%	79%

These results show no trend toward improved performance when using more ROI filters (and proportionally more correlations). In interpreting these results, it must be remembered that the test images contain the target at only three specific poses while the filter sets are designed to cover 360 degrees of azimuth and a range variation of 1.6:1. The effect of this is unknown.

Significant improvements in return for using more clusters can be found with more scrutiny of the results. For the four-cluster case, no filter detected the ROI as the top correlation peak for two of the inputs containing the tank at 180 degrees azimuth (i.e., an end-on view). The ROI was detected as the second-ranked peak in these two cases. With 13 cluster-ROI's, first-rank ROI detection was obtained for one filter for these same two cases.

The average ROI correlation peak intensity increased from 4.73 units with four clusters to 5.20 units with 13 clusters, and the average clutter rejection did not change.

Overall, the increase in performance seems slight compared with the large increase in the number of correlations that must be performed and analyzed.

6.6 Clustering into 114 Clusters

When the clustering program was given the goal of 120 clusters it defined 114 clusters. The cardinal poses for these of course correspond to 114 of the 120 poses included in the distance matrix. There were two clusters each of two and three poses and 110 clusters consisting only of a cardinal pose. This was a good test that the cluster program could handle the extreme case and provide sensible results.

6.7 Cluster Interpretation

One might expect that a rational clustering into a sufficient number of clusters would group training image poses in an intuitive way, i.e., that neighboring azimuth and range values would tend to be clustered together. This expectation may be unreasonable when only four clusters are used to span the huge distortion space involved in this test case, as is indicated in Figure 2, which is a scatter plot of the aspects clustered in the first of the four defined clusters.

Figure 3 plots the aspects clustered in the fourth cluster for the 13-cluster case. The degree to which this cluster grouping matches intuitive expectations is arguable, and the reader is left to draw their own conclusions. These plots are representative of the remaining plots not included herein. It should be remembered that the distortion space assigned to each of the 120 training views exceeds the distortion coverage assigned to a typical smart filter.

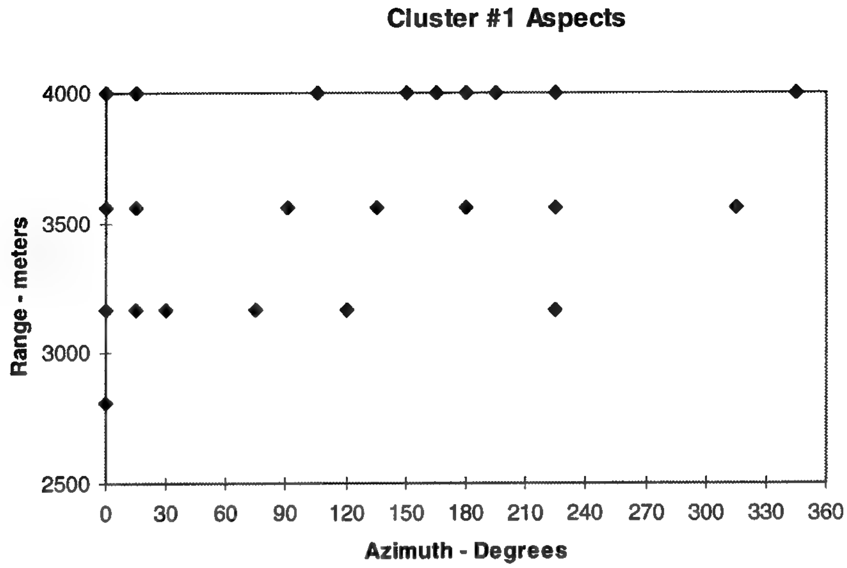

Figure 2. Aspects in first cluster of four-cluster case.

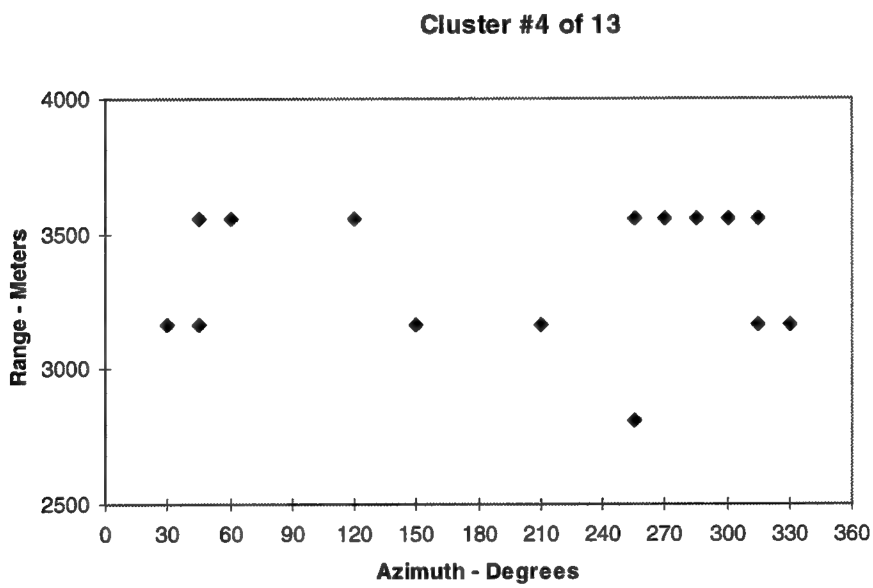

Figure 3. Aspects in fourth cluster of 13-cluster case.

7.0 CONCLUSION AND RECOMMENDATIONS

The work reported herein supports the following conclusions:

1. Symmetric and non-symmetric normalized BPOF correlations can be defined. They serve as the basis for defining distance (or proximity) metrics that are useful for clustering training images and for improving the response uniformity of LASE filter formulations via the iterated-LASE (ILASE) formulation.
2. A practical clustering algorithm using the symmetric normalized BPOF distance metric has been implemented in software and tested.
3. The ILASE filter formulation has been successfully implemented in software.
4. Effective region-of-interest BPOF correlation filters were formulated based on normalized-BPOF clusters using the ILASE formulation. These filters were shown to be far superior to another accepted type of ROI filter in correlation simulations using realistic input scenes.

The following are recommendations for further study:

1. The cluster-ROI filters should be tested on an actual application.
2. The cluster-ROI concept should be extended to other filter modulation schemes, including phase-only filters.
3. The potential for cluster-based filter formulations to provide superior filter sets for decision-tree correlation recognition strategies, and for improved smart filters in general, should be explored in further research studies.

8.0 REFERENCES

1. S.D. Lindell, "Summary of The Transfer of Optical Processing to Systems: Optical Pattern Recognition Program," Proc. SPIE, Vol. 2489, p. 20, Orlando, Florida, 18 April 1995.
2. D. L. Flannery, "Optimal trade off distortion tolerant constrained modulation correlation filters," JOSA-A, Vol. 12, p. 66, January 1995.
3. D. L. Flannery, "Correlation Filter and Preprocessing Algorithms for LCIR Images," UDR-TR-95-23, March 1995
4. D. Flannery, J. Loomis and M. Milkovich, "Design elements of binary phase-only correlation filters," Applied Optics, Vol. 27, p. 4231, 15 October 1988.
5. D.L. Flannery and W.B. Hahn, Jr., "TOPS pattern recognition Algorithms and their extension to the IR," Proc. SPIE Vol. 2489, p. 9, Orlando, 18 April 1995.
6. D. Jared and D. Ennis, "Inclusion of filter modulation in synthetic-discriminant function filters," Applied Optics, Vol. 28, p. 232, 15 January 1989.
7. D. Flannery, "Optical Correlation Studies," UDR-TR-96-45, May 1996.

☆U.S. GOVERNMENT PRINTING OFFICE: 1997-509-127-47080

DISTRIBUTION LIST

addresses	number of copies
JAMES L. DAVIS ROME LABORATORY/OCPC 25 ELECTRONIC PKY ROME NY 13441-4515	5
DAVID L. FLANNERY UNIVERSITY OF DAYTON RESEARCH INST 300 COLLEGE PARK DAYTON OH 45469-0150	5
ROME LABORATORY/SUL TECHNICAL LIBRARY 26 ELECTRONIC PKY ROME NY 13441-4514	1
ATTENTION: DTIC-OCC DEFENSE TECHNICAL INFO CENTER 8725 JOHN J. KINGMAN ROAD, STE 0944 FT. BELVOIR, VA 22060-6218	2
ADVANCED RESEARCH PROJECTS AGENCY 3701 NORTH FAIRFAX DRIVE ARLINGTON VA 22203-1714	1
ATTN: RAYMOND TADROS GIDEP P.O. BOX 8000 CORONA CA 91718-8000	1
ATTN: WALTER HARTMAN WRIGHT LABORATORY/AAM, BLDG. 620 2241 AVIONICS CIRCLE, RM N3-F10 WRIGHT-PATTERSON AFB OH 45433-7333	1
AFIT ACADEMIC LIBRARY/LDEE 2950 P STREET AREA B, BLDG 642 WRIGHT-PATTERSON AFB OH 45433-7765	1

ATTN: R.L. DENISON 1
WRIGHT LABORATORY/MLPO, BLDG. 651
3005 P STREET, STE 6
WRIGHT-PATTERSON AFB OH 45433-7707

WRIGHT LABORATORY/FIVS/SURVIAC 1
2130 EIGHTH STREET, BLDG 45, STE 1
WRIGHT-PATTERSON AFB OH 45433-7542

ATTN: GILBERT G. KUPERMAN 1
AL/CFHI, BLDG. 248
2255 H STREET
WRIGHT-PATTERSON AFB OH 45433-7022

OL AL HSC/HRG, BLDG. 190 1
2698 G STREET
WRIGHT-PATTERSON AFB OH 45433-7604

AUL/LSAD 1
600 CHENNAULT CIRCLE, BLDG. 1405
MAXWELL AFB AL 36112-6424

US ARMY STRATEGIC DEFENSE COMMAND 1
CSSD-IM-PA
P.O. BOX 1500
HUNTSVILLE AL 35807-3801

NAVAL AIR WARFARE CENTER 1
6000 E. 21ST STREET
INDIANAPOLIS IN 46219-2189

COMMANDING OFFICER 1
NCCOSC RDT&E DIVISION
ATTN: TECHNICAL LIBRARY,CODE 0274
53560 HULL STREET
SAN DIEGO CA 92152-5001

COMMANDER, TECHNICAL LIBRARY 1
4747000/C0223
NAVAIRWARCENWPNDIV
1 ADMINISTRATION CIRCLE
CHINA LAKE CA 93555-6001

SPACE & NAVAL WARFARE SYSTEMS COMMAND (PMW 178-1) 2451 CRYSTAL DRIVE ARLINGTON VA 22245-5200	2
SPACE & NAVAL WARFARE SYSTEMS COMMAND, EXECUTIVE DIRECTOR (PD13A) ATTN: MR. CARL ANDRIANI 2451 CRYSTAL DRIVE ARLINGTON VA 22245-5200	1
COMMANDER, SPACE & NAVAL WARFARE SYSTEMS COMMAND (CODE 32) 2451 CRYSTAL DRIVE ARLINGTON VA 22245-5200	1
CDR, US ARMY MISSILE COMMAND RSIC, BLDG. 4484 AMSMI-RD-CS-R, DOCS REDSTONE ARSENAL AL 35898-5241	2
ADVISORY GROUP ON ELECTRON DEVICES SUITE 500 1745 JEFFERSON DAVIS HIGHWAY ARLINGTON VA 22202	1
REPORT COLLECTION, CIC-14 MS P364 LOS ALAMOS NATIONAL LABORATORY LOS ALAMOS NM 87545	1
AEDC LIBRARY TECHNICAL REPORTS FILE 100 KINDEL DRIVE, SUITE C211 ARNOLD AFB TN 37389-3211	1
COMMANDER USAISC ASHC-IMD-L, BLDG 61801 FT HUACHUCA AZ 85613-5000	1
AFIWC/MSO 102 HALL BLVD, STE 315 SAN ANTONIO TX 78243-7016	1

NSA/CSS K1 FT MEADE MD 20755-6000	1
PHILLIPS LABORATORY PL/TL (LIBRARY) 5 WRIGHT STREET HANSCOM AFB MA 01731-3004	1
THE MITRE CORPORATION ATTN: E. LADURE D460 202 BURLINGTON RD BEDFORD MA 01732	1
OUSD(P)/DTSA/DUTD ATTN: PATRICK G. SULLIVAN, JR. 400 ARMY NAVY DRIVE SUITE 300 ARLINGTON VA 22202	2
ROME LABORATORY/ERO ATTN: RICHARD PAYNE HANSCOM AFB, MA 01731-5000	1
ROME LABORATORY/EROC ATTN: JOSEPH P. LORENZO, JR. HANSCOM AFB, MA 01731-5000	1
ROME LABORATORY/EROP ATTN: JOSEPH L. HORNER HANSCOM AFB, MA 01731-5000	1
ROME LABORATORY/EROC ATTN: RICHARD A. SOREF HANSCOM AFB, MA 01731-5000	1

ROME LABORATORY/ERXE 1
ATTN: JOHN J. LARKIN
HANSCOM AFB, MA 01731-5000

ROME LABORATORY/ERDR 1
ATTN: DANIEL J. BURNS
525 BROOKS RD
ROME NY 13441-4505

ROME LABORATORY/IRAP 1
ATTN: ALBERT A. JAMBERDINO
32 HANGAR RD
ROME NY 13441-4114

ROME LABORATORY/OCP 1
ATTN: BRIAN M. HENDRICKSON
25 ELECTRONIC PKY
ROME NY 13441-4515

ROME LABORATORY/OCPC 1
ATTN: GREGORY J. ZAGAR
25 ELECTRONIC PKY
ROME NY 13441-4515

ROME LABORATORY/C3BC 1
ATTN: ROBERT L. KAMINSKI
525 BROOKS RD
ROME NY 13441-4505

ROME LABORATORY/OCP 1
ATTN: JAMES W. CUSACK
25 ELECTRONIC PKY
ROME NY 13441-4515

ROME LABORATORY/OCP 1
ATTN: JOANNE L. ROSSI
25 ELECTRONIC PKY
ROME NY 13441-4515

ROME LABORATORY/OCPA 1
ATTN: ANDREW R. PIRICH
25 ELECTRONIC PKY
ROME NY 13441-4515

ROME LABORATORY/OCP 1
ATTN: RICHARD J. MICHALAK
25 ELECTRONIC PKY
ROME NY 13441-4515

NY PHOTONIC DEVELOPMENT CORP 1
MVCC ROME CAMPUS
UPPER FLOYD AVE
ROME, NY 13440

NRAD/CODE 55 1
ATTN: DR. STEVE PAPPERT
SAN DIEGO CA 92152-5000

PHILLIPS LABORATORY/VTET 2
ATTN: MR. EDWARD W. TAYLOR
3550 ABERDEEN AVENUE SE, BLDG 902
KIRTLAND AFB NM 87117-5776

LABORATORY FOR PHYSICAL SCIENCES 1
ATTN: DONALD LA FAW
4928 COLLEGE AVENUE
COLLEGE PARK MD 20740

WRIGHT LABORATORY/XPN 3
ATTN: E. UTT
BLDG 254
2591 "K" STREET
WRIGHT-PATTERSON AFB OH 45433-7602

USA PHOTONICS RESEARCH CELL 2
ATTN: LTC A. H. SAYLES
DEPT OF EE AND CS
UNITED STATES MILITARY ACADEMY
WEST POINT, NY 10996

USA/CECOM 2
ATTN: AMSEL-RD-C3-AC-L
ATTN: J. WRIGHT/L. CORYELL
FORT MONMOUTH NJ 07703-5205

NRAD/CODE 55 1
ATTN: DR. STEVE PAPPERT
SAN DIEGO CA 92152-5000

MISSION

OF

ROME LABORATORY

Mission. The mission of Rome Laboratory is to advance the science and technologies of command, control, communications and intelligence and to transition them into systems to meet customer needs. To achieve this, Rome Lab:

a. Conducts vigorous research, development and test programs in all applicable technologies;

b. Transitions technology to current and future systems to improve operational capability, readiness, and supportability;

c. Provides a full range of technical support to Air Force Materiel Command product centers and other Air Force organizations;

d. Promotes transfer of technology to the private sector;

e. Maintains leading edge technological expertise in the areas of surveillance, communications, command and control, intelligence, reliability science, electro-magnetic technology, photonics, signal processing, and computational science.

The thrust areas of technical competence include: Surveillance, Communications, Command and Control, Intelligence, Signal Processing, Computer Science and Technology, Electromagnetic Technology, Photonics and Reliability Sciences.